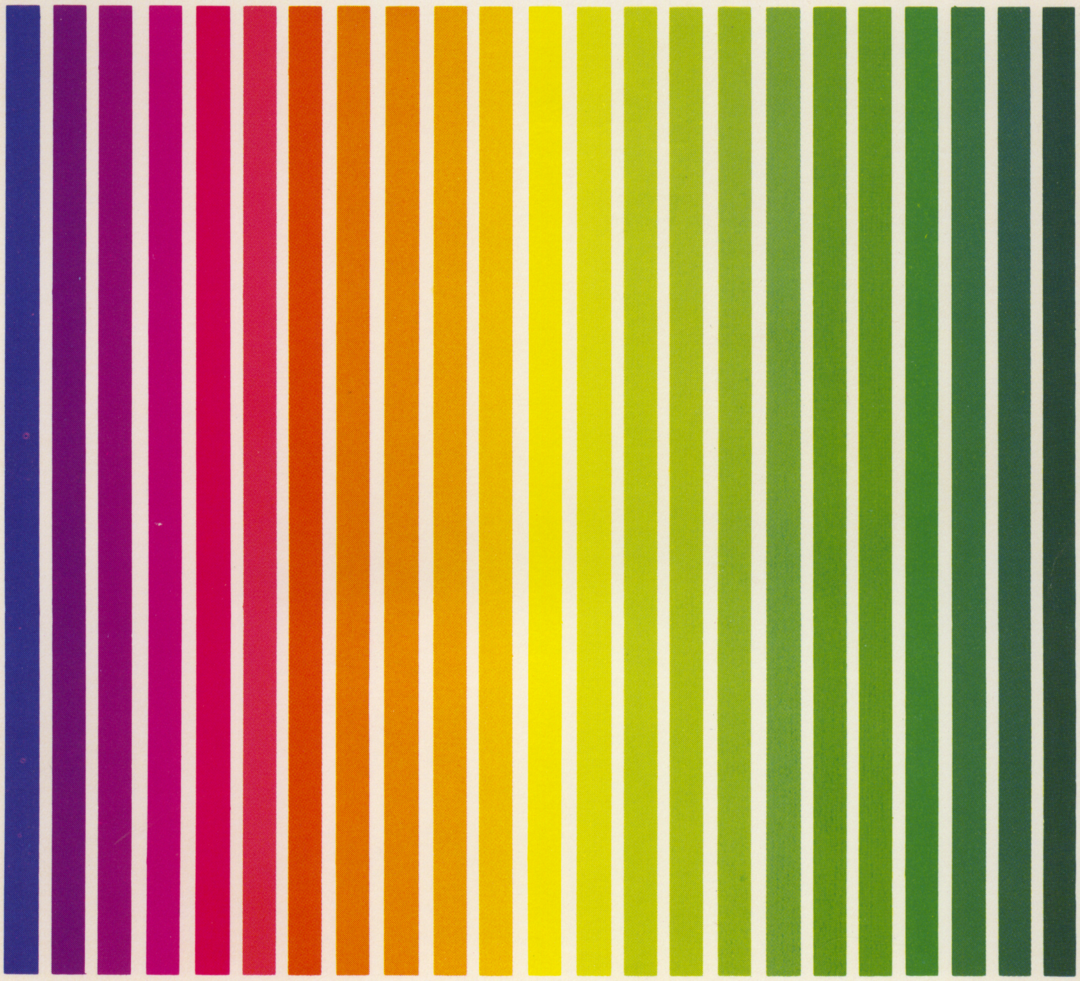
ATARI® PROGRAM EXCHANGE



Mike Ekberg

DISK FIXER (FIX) , REV 2.0

APX-20010

User-Written Software for ATARI Home Computers

Mike Ekberg

DISK FIXER (FIX) , REV 2.0

APX-20010

DISK FIXER, Rev. 2.0
(FIX)

by

Michael Ekberg

Program and Manual Contents © 1981 Atari, Inc.

Copyright and right to make backup copies. On receipt of this computer program and associated documentation (the software), ATARI grants to you a nonexclusive license to execute the enclosed software and to make backup or archival copies for your personal use only, and only on the condition that all copies are conspicuously marked with the same copyright notices as appear on the original. This software is copyrighted. You are prohibited from reproducing, translating, or distributing this software in any unauthorized manner.

TRADEMARKS OF ATARI

The following are trademarks of Atari, Inc.

ATARI
ATARI 400 Home Computer
ATARI 800 Home Computer
ATARI 410 Program Recorder
ATARI 810 Disk Drive
ATARI 820 40-Column Printer
ATARI 822 Thermal Printer
ATARI 825 80-Column Printer
ATARI 830 Acoustic Modem
ATARI 850 Interface Module

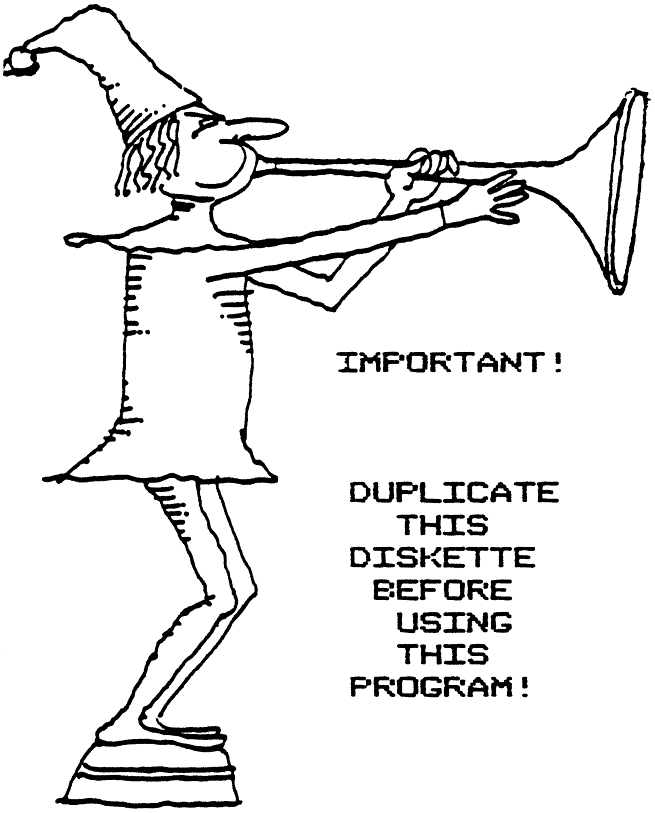
****
Distributed by

The ATARI Program Exchange
P. O. Box 427
155 Moffett Park Drive, B-1
Sunnyvale, CA 94086

To request an APX Software Catalog, write to the address above, or call toll-free:

800/538-1862 (outside California)
800/672-1850 (within California)

Or call our Sales number, 408/745-5535.
****



IMPORTANT!

DUPLICATE
THIS
DISKETTE
BEFORE
USING
THIS
PROGRAM!

This APX diskette is unnotched to protect the software against accidental erasure. However, this protection also prevents a program from storing information on the diskette. The program you've purchased involves storing information. Therefore, before you can use the program, you must duplicate the contents of the diskette onto a notched diskette that doesn't have a write-protect tab covering the notch.

To duplicate the diskette, call the Disk Operating System (DOS) menu and select option J, Duplicate Disk. You can use this option with a single disk drive by manually swapping source (the APX diskette) and destination (a notched diskette) until the duplication process is complete. You can also use this option with multiple disk drive systems by inserting source and destination diskettes in two separate drives and letting the duplication process proceed automatically. (Note. This option copies sector by sector. Therefore, when the duplication is complete, any files previously stored on the destination diskette will have been destroyed.)

CONTENTS

PREFACE

- Overview
- Required accessories
- Getting started

INTRODUCTION TO FIX __ 1

FILE STRUCTURE __ 1

MENU ITEMS __ 2

- A: Directory Entries __ 2
- B: Trace Sector Chain __ 2
- C: Modify Directory Entry __ 3
- D: Check Allocation Map __ 4
- E: Modify Sector Link __ 4
- F: Set Drive Number __ 5
- G: Exit to DOS __ 5
- H: Dump Sectors __ 5
- I: Edit Sector __ 6

ERROR CODES AND RECOVERY __ 7

- 128 BREAK KEY ABORT __ 7
- 129 IOCB ALREADY OPEN __ 7
- 130 NON-EXISTENT DEVICE __ 7
- 131 IOCB WRITE ONLY ERROR __ 7
- 132 ILLEGAL HANDLER COMMAND __ 7
- 133 DEVICE/FILE NOT OPEN __ 7
- 134 INVALID IOCB NUMBER __ 7
- 135 IOCB WRITE ONLY ERROR __ 7
- 136 END OF FILE __ 8
- 137 TRUNCATED RECORD __ 8
- 138 DEVICE TIMEOUT __ 8
- 139 DEVICE NAK __ 9
- 140 SERIAL FRAME ERROR __ 9
- 141 CURSOR OUTRANGE __ 9
- 142 SERIAL OVERRUN __ 9
- 143 CHECKSUM ERROR __ 9
- 144 DEVICE DONE ERROR __ 9
- 145 READ AFTER WRITE COMPARE ERROR __ 10
- 146 FUNCTION NOT IMPLEMENTED __ 10
- 147 NOT ENOUGH RAM FOR SELECTED GRAPHICS MODE __ 10
- 160 DRIVE NUMER ERROR __ 10
- 161 TOO MANY OPEN FILES __ 10
- 162 DISK FULL __ 10

163 FATAL SYSTEM I/O ERROR __ 10
164 FILE NUMBER MISMATCH __ 10
165 FILENAME ERROR __ 10
166 POINT DATA LENGTH __ 11
167 FILE LOCKED __ 11
168 DEVICE COMMAND INVALID __ 11
169 DIRECTORY FULL __ 11
170 FILE NOT FOUND __ 11
171 POINT INVALID __ 11
172 ILLEGAL APPEND __ 11
173 BAD SECTORS AT FORMAT __ 11

FIGURES 1 - 7 __ 12 - 16

PREFACE

OVERVIEW

FIX lets an advanced programmer get directly at several important areas of a diskette. Depending on the circumstances, FIX can sometimes help you recover some or all of your files from a "crashed" diskette. FIX lets you: (1) display directory entries, so that you know the exact entry for each file; (2) modify directory entries, so that you can control directory information; (3) trace sector chains through a file, looking for the end of the file or for a bad sector, so that you can verify the structure of a file; (4) check the allocation map, so that you can recover misallocated sectors; (5) modify sector links, so that you can control the file number, sector byte count, and forward sector pointer for any sector on the diskette and (6) edit actual sector data.

This utility program is for an advanced systems programmer only. Its use requires a detailed understanding of disk structure.

REQUIRED ACCESSORIES

24K RAM
ATARI 810 Disk Drive

GETTING STARTED

1. Turn on your disk drive, insert the DISK FIXER diskette, and power up your computer.

2. When the READY prompt displays, type DOS to call up the menu.

3. Enter menu selection L (Binary Load).

4. To the LOAD FROM WHAT FILE? prompt, enter FIXDMP and press the RETURN key. The file will load into RAM and start.

I. INTRODUCTION TO FIX

FIX is an assembly language utility to examine and modify disk files. All numbers output by and entered to FIX are hexidecimal. You can use the CONTROL-1 key to start and stop the data display on the screen.

II. FILE STRUCTURE

Diskettes are divided into blocks of data called sectors. Diskettes for the ATARI 810 Disk Drive have 720 sectors of 128 bytes. Diskettes for the ATARI 815 Dual Disk Drive have 720 sectors of 256 bytes. Diskettes have four parts:

1. BIT MAP. The bit map contains information on the status of each sector. By examining the bit map, the File Manager can determine if a sector has been allocated (in use) or not (free).

2. DIRECTORY. The directory sectors contain information on the names, location, length, and status of the files on a diskette.

3. BOOT SECTORS. Each diskette contains one or three sectors reserved for information read on power-up. These sectors are normally inaccessible to the user when using the File Manager. All 9/24/79 DOS diskettes use sector one for the boot. All DOS II 2S or 2D diskettes use sectors one through three.

4. DATA SECTORS. The rest of the sectors are used to store the information in the files. Most of the bytes of a sector are dedicated to the actual file information. The last three bytes of every sector contain housekeeping information for the File Manager.

FILES are made up of a DIRECTORY ENTRY and a set of one or more sectors. The sectors are usually contiguous, but do not have to be. For a complete description of a directory entry, see section III.A.

III. MENU ITEMS

FIX is a menu-driven program. To execute commands, type the letter in front of the command listed in the displayed menu (see Figure 1). You needn't press the RETURN key; FIX automatically accepts your command, and it replies with the prompt for that command. If you inadvertently type the wrong command, press the BREAK key to abort the command.

A: Directory Entries

This menu item displays the entries in the directory. Figure 2 shows the user requesting FIX to display the first eight (hex) entries. Notice that the entries start with zero. The directory may contain a maximum of 64 files. Single entries may be displayed by simply typing the file number and pressing the RETURN key.

The leftmost column gives the file number of each file displayed. The next column contains the file name and extender. Notice that the period is omitted in the file name. The next column, labeled FSEC, is the first sector in the file. The #SEC column lists the number of sectors in the file. Both numbers are hexidecimal.

The last column, DL, notes the status of the file. Notice in Figure 2 that file 4 (FIX6) has a D, meaning the file has been deleted from the directory. Because it is the first available "hole", the next file created will occupy its directory entry. File 3 (MYPROG.BAS) has an L, indicating the file is locked.

Files 5 - 8 contain no directory entries.

B: Trace Sector Chain

This menu item verifies the structure of a file. It traces the file through all its sectors until it hits an end-of-file or a bad link. A link is a pointer to the next sector of a file. A bad link is a link to a sector belonging to another file. An end-of-file is a link pointing to sector 0.

Figure 3 shows the sector chain for the small BASIC file MYPROG.BAS. Notice we're tracing file number 3. The file number is the entry number into the directory.

Column BS has two different meanings, depending on the DOS that created the file. The 9/24/79 DOS A used this number as the relative sector number. For the first sector in a file, this should be 1. Subsequent sectors should add 1 to this number. If you notice that numbers are skipped, this may indicate that the links in your file might be bad. The last sector's BS is the number of bytes in the sector.

For DOS II, the BS indicates the number of bytes in each sector. Most sectors contain $7D bytes (128-byte, single-density diskette) or $FD bytes (256-byte, double-density diskette). The last sector in a file is usually less than the above numbers.

Appending one file to another will cause the byte count in the sector between the appended portions to be less than full. It is therefore possible to have partially filled sectors in the middle of a file.

Column FP is the forward pointer. This is the link to the next sector of the file. Files on newly formatted diskettes tend to be linked sequentially. Files on diskettes with several files that have been deleted tend to become fractured, meaning that the files tend not to be linked sequentially. In other words, the sectors are not physically continguous.

C: Modify Directory Entry

Use this menu item to make changes to the diskette's directory. Enter the file number of the entry you wish to change and press the RETURN key.

Figure 4 shows the user is making a change to directory entry 3. FIX has printed the directory item and positioned the cursor to the file name.

At this point, you can edit the file name, the starting sector number, the number of sectors, or the status of the file. By inserting or deleting the D, you can delete or reinstate the file. By adding or deleting the L, you can lock or unlock the file. See section III.A for the meaning of the rest of the information on the line.

If you wish to have FIX update the entry, then press the RETURN key. If you don't want the entry updated, then press the BREAK key.

D: Check Allocation Map

This menu item can recover sectors that have become misallocated. To run it, type D. FIX automatically starts making its own bit map. It will link through all the files and mark all the sectors in use and those that are free. FIX then compares its bit map to the bit map on the diskette. All sectors that should be free but aren't are printed as "MARKED IN USE". All sectors that are really allocated to a file but aren't in the diskette's bit map are marked "FREE". Sectors that are correctly allocated aren't printed.

After FIX has checked all the sectors, it will exit back to the menu if all sectors are correct. If some sectors are misallocated, it will ask you if you want to write its correct copy to the diskette. Type Y to do so. If you don't want the corrected bit map to be written to the disk, then press the RETURN key.

Figure 5 shows the result of a D command. Notice that we had a bad link in file 0 (the first file). This will usually cause an error code 164 to display when you attempt to read the file. We would probably want to trace its sector chain (B command) to find the bad link. Our bit map seems also to be slightly mangled, so we would type Y to rewrite our new bit map.

E: Modify Sector Link

You may have discovered that some of your sector links have been altered incorrectly. This menu item allows you to change the links in the sectors on the diskette. After you type E , FIX prompts you to enter the sector number to modify. Enter the hex sector number and press the RETURN key.

FIX then prints the sector link information from the sector on the diskette. FIX positions the cursor on the line. Using the cursor control keys, you can alter the file number this sector is allocated to, the number of bytes in this sector of the file, and the forward pointer. Typing an E to the right of the forward pointer will mark this sector as end-of-file.

To tell FIX to write out the updated information to the sector, press the RETURN key. If you realize you've made an error in trying to modify this sector, first position the cursor below the line and then press the RETURN key. FIX will then redisplay the line. Press the RETURN key to exit.
See section III.B for more information on the sector link display.

F: Set Drive Number

This menu item sets the disk drive number. Type the disk drive number (1 - 8) and press the RETURN key. Figure 7 shows a user setting the disk drive number to 1 (FIX defaults to disk drive 1 upon loading).

G: Exit to DOS

This item causes a return to DOS.

H: Dump Sectors

This item lets you examine sectors on the diskette. To look at a single sector, type the sector number (in hex) and press the RETURN key. To look at several contiguous sectors, type the starting sector, a comma, and then the ending sector; then press the RETURN key. As the information scrolls up the screen, use CTRL-1 to stop and resume the display (that is, hold down the CTRL key while pressing 1).

Figure 8 shows a typical dump of a 128-byte sector. The sector links for this sector are at the top of the screen (see section III.B). The bytes in the sector are displayed in an 8-by-16 matrix in both hex and ATASCII. The leftmost column is the position of the next column's byte in the sector.

I: Edit Sector

This item lets you examine and modify a single sector on the diskette. In response to the prompt, enter the sector number (in hex) and press the RETURN key. FIX will then display a full 128-byte sector or the first half of a 256-byte sector similarly to the H option. To edit the sector, position the cursor on the line to be modified. Notice that in the left-hand column the hex position of the byte in the next column is displayed. Use these numbers to guide you in editing lines. You can edit only the hex bytes in the sector display. The ASCII characters are for reference only.

After editing a line, press the RETURN key. FIX will then redisplay the sector (or sector part) with your modifications. After you've finished editing, press the BREAK key. FIX will ask you if you wish to write the modified sector to the diskette. To do so, press the Y key. Press any other key to cause FIX to ask you if you wish to exit. Press the Y key to return to the menu without changing this sector on the diskette. Or, to cause FIX to ask you whether you want to write this sector to another sector, press any key other than Y. Then type in the number of the sector to which you want FIX to write this sector. Or, press the RETURN key to redisplay the prompt to exit.

Editing a 256-byte sector differs slightly from editing a 128-byte sector, described above. FIX displays the first half of the sector (bytes 0 - 7F hex). You edit this part as described. Pressing the BREAK key after editing the first part of the sector lets you edit the second part (bytes 80 - FF). You then press the BREAK key again for FIX to ask you if you want to write the sector, exit, and so on.

Figure 9 shows the editing of sector 5. The display resembles that for the H command. We modify sector 5 and then press the BREAK key. We respond to the two prompts by pressing the RETURN key, because we wish to write our edited sector to sector 8. We then type 8 <RETURN> and FIX writes the edited sector to sector 8.

IV. ERROR CODES AND RECOVERY

ERROR # DESCRIPTION

****

128 BREAK KEY ABORT
This is caused by pressing the BREAK key.

129 IOCB ALREADY OPEN
The IOCB is already open.
Recovery. You can CLOSE the IOCB and reOPEN it. You can use the current IOCB. You may want to check to see why the IOCB is OPEN.

130 NON EXISTENT DEVICE
You have tried to access a device not in the handler table; i.e., the device is undefined. The handler may be loaded in memory but not initialized. This error may occur when trying to access the ATARI 850 Interface Module without running the RS232 AUTORUN.SYS file.
Recovery. Check your I/O command for the correct device, or load and initialize the correct handler.

131 IOCB WRITE ONLY ERROR
You have attempted to read from a file opened for write only.
Recovery. Open the file for read or read/write (update mode).

132 ILLEGAL HANDLER COMMAND
This is a CIO error code. The command code passed to the device handler is illegal. The command is either less than or equal to 2 or is a special command to a handler that hasn't implemented any special commands.
Recovery. Check your XIO or IOCB command code for illegal code.

133 DEVICE/FILE NOT OPEN
Device or file not open. You haven't OPENed this file or device.
Recovery. Check your OPEN statement or file I/O statement for the wrong file specification.

134 INVALID IOCB NUMBER
You've tried to use an illegal IOCB index. For BASIC, the range is 1 - 7. BASIC doesn't allow use of IOCB 0. The Assembler Editor Cartridge requires the IOCB index to be a multiple of 16 and to be less than 128.

135 IOCB WRITE ONLY ERROR
You've tried to write to a device/file that is OPEN for read only. Recovery. You could open the file for write or for read/write.

136 END OF FILE

Your input file is at end-of-file. No more data is in the file.

137 TRUNCATED RECORD
This error typically occurs when the record you're reading is larger than the maximum record size specified in the call to CIO. BASIC's maximum record size is 119 bytes.
Recovery. You probably are trying to use an INPUT (record-oriented) command on a file that was created using PUT (byte-oriented) commands.

Records in a file are delimited by EOL (End Of Line) characters ($9B). Files created using PUTs have no EOL unless the bytes you output have a $9B. Trying to read a PUT file with record I/O may cause the DOS to read a record the size of the file. The DOS will then generate this error. Try reading the file using PUT statements.

This error may also occur when ENTERing a BASIC program created using SAVE. Try LOADing the program.

138 DEVICE TIMEOUT
A command was sent by the computer to a device over the serial bus. The device didn't respond within the period set by the Operating System (OS) for that particular device command. This error can be caused in a couple of ways.

1. The device number may be wrong. For example, the command

```
OPEN #3,4,0,"D3:MYPROG"
```

could cause this error to display if disk drive 3 isn't connected to the computer, turned on, or present.

2. The disk drive may be present but is unable to execute the command in the proper period of time.

Recovery. To recover from error case 1, examine the connections between the disk drive and the computer to make sure they're properly secured. Check the drive to make sure it's powered on and set for the correct drive number. Check your I/O command for the correct drive number. Retry the command. If you get this error again, try the recovery technique for the second error case. The second type of timeout error is caused by an interaction between the OS and the disk drive. DOS 9/24/79's timeout value was set too low for all possible cases of disk operations. The disk may have had some trouble executing the command in the short period allowed by the operating system. DOS II doesn't have this problem. This type of timeout error is intermittent. You should be able to retry the operation and succeed. If this error occurs more than a few times, the device may need repair.

139 DEVICE NAK
This is a catch-all error code. The device may have received a valid command, but can't execute it because of bad parameters--for example, trying to read an unaddressable sector, such as sector 0. Or, the device may have received a garbled command or data frame from the computer.
Recovery. Check your I/O command for illegal parameters.

140 SERIAL FRAME ERROR
Bit 7 of SKSTAT in POKEY is set. This means communications from the device to the computer are garbled. Specifically, POKEY detected missing or extra bits in a byte received on the serial data bus.

141 CURSOR OUTRANGE
Your cursor is out of range for this particular graphics mode.

142 SERIAL OVERRUN
Bit 5 of SKSTAT in POKEY is set. The computer didn't respond fast enough to a serial bus input interrupt. POKEY received another 8-bit word on the serial bus before the computer could process the previous word received.
Recovery. This is a rare error. If it recurs, you should have your computer serviced.

143 CHECKSUM ERROR
Serial bus communications are confused. The checksum sent by the device isn't the same as that calculated from the frame received by the computer.
Recovery. You can't do much about this. It could be a hardware or a software problem in the device/computer.

144 DEVICE DONE ERROR
The device can't execute a valid command. This error has two causes. Usually it means you've tried to write to a write-protected device or diskette. This is easily corrected by removing the diskette protect tab or by turning off the write-protect switch on the ATARI 815.

The second reason for this error appears unknown at this point. The disk drive is unable to read/write the sector requested or the cassette is set at the wrong baud rate.
Recovery. Remove the write-protect tab or turn off the write-protect switch. If the diskette wasn't write-protected, you might have problems with the diskette media. (I don't know how to recover on cassette.)

145 READ AFTER WRITE COMPARE ERROR
You've tried to open the SCREEN EDITOR with an illegal graphics mode number.
Recovery. Check GRAPHICS mode call or the AUX2 byte in the IOCB.

146 FUNCTION NOT IMPLEMENTED
Function not implemented in handler, e.g., trying to PUT to the keyboard or issuing special commands to the keyboard.
Recovery. Check your I/O command for the right command to the correct device.

147 NOT ENOUGH RAM FOR SELECTED GRAPHICS MODE
You don't have sufficient RAM installed for the selected graphics mode.
Recovery. Add more memory or use a smaller graphics mode.

160 DRIVE NUMBER ERROR
You specified an out-of-range drive (not 1 - 8) or you haven't allocated a buffer for this drive.
Recovery. Check your file specification or byte 1802 ($710).

161 TOO MANY OPEN FILES
You don't have any free sector buffers to use on another file.
Recovery. You may have up to 8 open files. Check location 1801 ($709) for the number of sector buffers allocated.

162 DISK FULL
You don't have any more free sectors on this diskette. This error occurs when writing to a full diskette.
Recovery. Use a different diskette with free sectors. Use the D option on FIX to see if you've misallocated sectors. At present, you can't recover from this error during program execution.

163 FATAL SYSTEM I/O ERROR
This error code means the File Manager has a bug in it. If you get this error, please report it to CUSTOMER SERVICE (800/538-8547, or 800/672-1430 for calls within California).

164 FILE NUMBER MISMATCH
The structure of the file is damaged. One of the file links points to a sector allocated to another file.
Recovery. Use B and E FIX commands to fix up the file links. You can't recover from this error during program execution.

165 FILENAME ERROR
Your file specification has illegal characters in it. Legal characters are alphanumerics, *, and ?.
Recovery. Check your file specification and remove illegal characters.

166 POINT DATA LENGTH

The byte count in the POINT call was greater than 125 (single-density) or 253 (double-density).

Recovery. Check POINT statement parameters.

167 FILE LOCKED

You've tried to access a locked file.

Recovery. Unlock the file.

168 DEVICE COMMAND INVALID

This probably means the device didn't recognize the command.

Recovery. ?????

169 DIRECTORY FULL

You don't have any free entries in the directory.

Recovery. See A and C FIX commands.

170 FILE NOT FOUND

You've tried to access a file that doesn't exist in the diskette's directory.

Recovery. Usually you've mistyped the file specification. Using the DOS A command, look at the directory for the correct name.

171 POINT INVALID

You've probably tried to POINT to a byte or record beyond the end of the file.

Recovery. Check the size of the file against the POINT value.

172 ILLEGAL APPEND

You've tried to OPEN a DOS I file for append using DOS II. DOS II can't append to DOS I files.

Recovery. Copy the DOS I file to a DOS II diskette, using DOS II.

173 BAD SECTORS AT FORMAT

The disk controller detected bad sectors while formatting a diskette.

Recovery. Throw away the diskette and use another. If you can't format a diskette, the disk drive might need repair.

```
FIXDUMP 2.0SD (C) ATARI 10/3/1980

A: DIRECTORY ENTRIES  H: DUMP SECTORS
B: TRACE SECTOR CHAIN  I: EDIT SECTOR
C:  MODIFY DIRECTORY ENTRY
D:  CHECK ALLOCATION MAP
E:  MODIFY SECTOR LINK
F:  SET DRIVE NUMBER
G:  EXIT TO DOS


SELECT ITEM OR RETURN FOR MENU
_
```

Figure 1. Disk Fixer Menu

```
SELECT ITEM OR RETURN FOR MENU
FIRST,LAST DIR ENTRIES TO SHOW?
0,8
D# FILENAM EXT FSEC #SEC DL
00 DOS     SYS 0004 0026
01 DUP     SYS 002A 002A
02 FIX6        0054 002A
03 MYPROG  BAS 007E 0003  L
04 FIX6        0054 0020 D
05 ########### 0000 0000
06 ########### 0000 0000
07 ########### 0000 0000
08 ########### 0000 0000

SELECT ITEM OR RETURN FOR MENU
_
```

Figure 2. Option A: Directory of Entries

```
SELECT ITEM OR RETURN FOR MENU
TRACE FROM WHAT DIR ENTRY?
3
03 MYPROG  BAS 007E 0003  L
SECTOR 07E: F#=03, BS=7D, FP=07F
SECTOR 07F: F#=03, BS=7D, FP=080
SECTOR 080: F#=03, BS=45, FP=000 E

SELECT ITEM OR RETURN FOR MENU
_
```

Figure 3. Option B: Trace Sector Chain

```
SELECT ITEM OR RETURN FOR MENU
WHICH ENTRY TO MODIFY?
3
D# FILENAM EXT FSEC #SEC DL
03 MYPROG  BAS 007E 0003  L
```

Figure 4. Option C: Modify Directory Entry

```
SELECT ITEM OR RETURN FOR MENU
BUILDING ALLOCATION MAP...
BAD LINK IN FILE # 00
SECTOR 001: F#=00, BS=0C, FP=203
WAS MARKED IN USE
SECTOR 004: F#=00, BS=7D, FP=005
WAS MARKED FREE
SECTOR 007: F#=00, BS=7D, FP=008
WAS MARKED IN USE
SECTOR 02A: F#=01, BS=7D, FP=02B
WAS MARKED FREE
SECTOR 02C: F#=01, BS=7D, FP=02D
WAS MARKED FREE
SECTOR 032: F#=01, BS=7D, FP=033
WAS MARKED FREE
```

Figure 5. Option D: Check Allocation Map

```
SELECT ITEM OR RETURN FOR MENU
MODIFY LINK OF WHAT SECTOR?
80
SECTOR 080: F#=03, BS=45, FP=000 E
```

Figure 6. Option E: Modify Sector Link

```
SELECT ITEM OF RETURN FOR MENU
USE WHAT DRIVE NUMBER?
1

SELECT ITEM OR RETURN FOR MENU
_
```

Figure 7. Option F: Set Drive Number

```
SECTOR 005: F#=00, BS=7D, FP=006
BYTE
00   10 01 98 9D 1A 13 98 30 .......0
08   0D A5 43 9D 3A 13 A5 44 ..C.:..D
10   9D 4A 13 20 70 08 E8 E0 .J. p...
18   10 D0 E2 A5 43 8D E7 02 ....C...

20   A5 44 8D E8 02 4C 7E 08 .D...L..
28   18 A5 43 69 80 85 43 A5 ..Ci..C.
30   44 69 00 85 44 60 A0 7F .DI..D#.
38   A9 00 99 82 13 88 D0 FA ........

40   A0 00 B9 1A 03 F0 0C C9 ........
48   44 F0 08 C8 C8 C8 C0 IE D.......
50   D0 F0 00 A9 44 99 1A 03 ....D...
58   A9 CB 99 1B 03 A9 07 99 ........

60   1C 03 60 20 65 11 20 98 ..# e. .
68   0E BD 4A 03 9D 83 13 29 ..J....)
70   02 F0 03 4C A7 0D 20 1B ...L.. .
78   0F 08 BD 83 13 00 06 7D ........

SELECT ITEM OR RETURN FOR MENU

_
```

Figure 2. Option H: Dump Sectors

```
SELECT ITEM OR RETURN FOR MENU
ENTER HEX DISK SECTOR TO EDIT
5
SECTOR 005: F#=00, BS=7D, FP=006
BYTE
00   10 01 98 9D 1A 13 98 30 .......0
08   0D A5 43 9D 3A 13 A5 44 ..C.:..D
10   9D 4A 13 20 70 08 E8 E0 .J. p...
18   10 D0 E2 A5 43 8D E7 02 ....C...

20   A5 44 8D E8 02 4C 7E 08 .D...L..
28   18 A5 43 69 80 85 43 A5 ..Ci..C.
30   44 69 00 85 44 60 A0 7F .DI..D#.
38   A9 00 99 82 13 88 D0 FA ........

40   A0 00 B9 1A 03 F0 0C C9 ........
48   44 F0 08 C8 C8 C8 C0 IE D.......
50   D0 F0 00 A9 44 99 1A 03 ....D...
58   A9 CB 99 1B 03 A9 07 99 ........

60   1C 03 60 20 65 11 20 98 ..# e. .
68   0E BD 4A 03 9D 83 13 29 ..J....)
70   02 F0 03 4C A7 0D 20 1B ...L.. .
78   0F 08 BD 83 13 00 06 7D ........

TYPE BRK TO EXIT

TYPE "Y" TO WRITE SECTOR
TYPE "Y" TO EXIT
ENTER HEX SECTOR TO WRITE BUFFER TO
8__
```

Figure 9. Option I: Edit Sector

DISCLAIMER OF WARRANTY AND LIABILITY ON COMPUTER PROGRAMS

Neither Atari, Inc. ("ATARI"), nor its software supplier, distributor, or dealers make any express or implied warranty of any kind with respect to this computer software program and/or material, including, but not limited to warranties of merchantability and fitness for a particular purpose. This computer program software and/or material is distributed solely on an "as is" basis. The entire risk as to the quality and performance of such programs is with the purchaser. Purchaser accepts and uses this computer program software and/or material upon his/her own inspection of the computer software program and/or material, without reliance upon any representation or description concerning the computer program software and/or material. Should the computer program software and/or material prove defective, purchaser and not ATARI, its software supplier, distributor, or dealer, assumes the entire cost of all necessary servicing, repair, or correction, and any incidental damages.

In no event shall ATARI, or its software supplier, distributor, or dealer be liable or responsible to a purchaser, customer, or any other person or entity with respect to any liability, loss, incidental or consequential damage caused or alleged to be caused, directly or indirectly, by the computer program software and/or material, whether defective or otherwise, even if they have been advised of the possibility of such liability, loss, or damage.

LIMITED WARRANTIES ON MEDIA AND HARDWARE ACCESSORIES

ATARI warrants to the original consumer purchaser that the media on which the computer software program and/or material is recorded, including computer program cassettes or diskettes, and all hardware accessories are free from defects in materials or workmanship for a period of 30 days from the date of purchase. If a defect covered by this limited warranty is discovered during this 30-day warranty period, ATARI will repair or replace the media or hardware accessories, at ATARI's option, provided the media or hardware accessories and proof of date of purchase are delivered or mailed, postage prepaid, to the ATARI Program Exchange.

This warranty shall not apply if the media or hardware accessories (1) have been misused or show signs of excessive wear, (2) have been damaged by playback equipment or by being used with any products not supplied by ATARI, or (3) if the purchaser causes or permits the media or hardware accessories to be serviced or modified by anyone other than an authorized ATARI Service Center. Any applicable implied warranties on media or hardware accessories, including warranties of merchantability and fitness, are hereby limited to 30 days from the date of purchase. Consequential or incidental damages resulting from a breach of any applicable express or implied warranties on media or hardware accessories are hereby excluded. Some states do not allow limitations on how long an implied warranty lasts, so the above limitation may not apply to you. Some states also do not allow the exclusion or limitation of incidental or consequential damage, so the above limitation or exclusion may not apply to you.

ATARI PROGRAM EXCHANGE

REVIEW FORM

We're interested in your experiences with APX programs and documentation, both favorable and unfavorable. Many software authors are willing and eager to improve their programs if they know what users want. And, of course, we want to know about any bugs that slipped by us, so that the software author can fix them. We also want to know whether our documentation is meeting your needs. You are our best source for suggesting improvements! Please help us by taking a moment to fill in this review sheet. Fold the sheet in thirds and seal it so that the address on the bottom of the back becomes the envelope front. Thank you for helping us!

1. Name and APX number of program ______________________________

2. If you have problems using the program, please describe them here.

____

____

____

3. What do you especially like about this program?

____

____

____

4. What do you think the program's weaknesses are?

____

____

____

5. How can the catalog description be more accurate and/or comprehensive?

____

____

6. On a scale of 1 to 10, 1 being "poor" and 10 being "excellent", please rate the following aspects of this program?

_____ Easy to use
_____ User-oriented (e.g., menus, prompts, clear language)
_____ Enjoyable
_____ Self-instructuve
_____ Useful (non-game software)
_____ Imaginative graphics and sound

7. Describe any technical errors you found in the user instructions (please give page numbers).

8. What did you especially like about the user instructions?

9. What revisions or additions would improve these instructions?

10. On a scale of 1 to 10, 1 representing "poor" and 10 representing "excellent", how would you rate the user instructions and why?

11. Other comments about the software or user instructions:

STAMP

ATARI Program Exchange
P.O. Box 427
155 Moffett Park Drive, B-1
Sunnyvale, CA 94086

[seal here]